JavaScript

JavaScript – is the programming language that runs in your browser. You can use it to add interactivity and other dynamic features to your website or application.

* Functions in javascript are like objects
  + Arrow syntax – fat arrow syntax; it is more compact
  + Var multiply = (a,b) – {return a\*b;}
  + Var triple = n – 3\*n;
  + Var zero = () = 0;
  + triple (false) – 0
* functions in javascript can be recursive
  + 0! = 1

If (n<0){

Throw “invalid argument”;

}else if (n==0){

Return 1;

} else{

Return n\*factorial(n-1);

}

* Function nest parameters
  + Function fn(a,b, …others){

.

.

.

}

* Arrays
  + Collection of elements/homogenous collection of elements
  + Dynamically sized
  + Length is unitable
  + Var array lengthline = new array(50
    - Elements are not yet created
    - Array with 5 slots
  + Array destructuring
    - var array = [1,2,3,4,5];
    - var [a,b,c,d,e] = array;

Core JS Object

* Array, Boolean, Date, Error, Funciton, JSON, Math, Number, Object, Reg. Exp. String, Map, Set WeakMap, WeakSet

Var instance of ‘type’ – tf

* n = 0/0 = NaN
* n = 1/0 = Infinity
* n -1/0 = Infinity

String prototype reverse = function{

Split(‘).reverse().join();

}

Standard Objects

Math – calls various mathematical objects

Functions

* void – doesn’t return values
* sayHello – without “()” – refers to the whole function with the name sayHello
* sayHello() – return value inside function
* parameter list – list names of parameters
* Inside Function
  + There are no void in javascript, all functions returns values
  + Condition not incuded in function will return ‘undefined’
  + Greater function in strings (lexicographic)
  + Lowercase is greater than uppercase letters
  + True, False \* greater = true
  + True, 12 = 12
  + a, 12 = undefined
  + automatic conversion is done when comparing variables identity operator
    - compare their types & values
  + functions with many arguments with different types (cas)
  + function expression (anonymous function)
    - case 5: return (function(a,b){return a+b;})
    - when enclosed in parenthesis it is converted into an expression
  + name conflict
    - using same names into different functions with different content